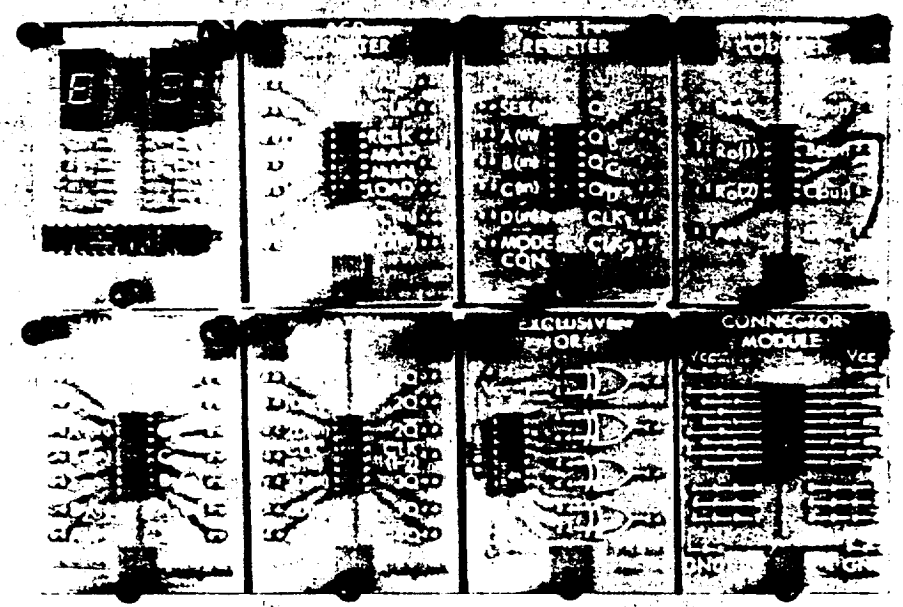
MSI FLEXIBILITY, EXPEDIENCY AND VERSATILITY

DIGITAL TRAINER OPTIONAL MODULES MODEL AA352

DV-6F23



OPTIONAL MODULES FOR MODEL AA352

GENERAL DESCRIPTION

The optional modules presently being offered for the Model AA352 Digital Trainer are TTL MSI integrated circuits that are fully compatible with the TTL SSI basic modules of the digital trainer. These modules are a **Dual Numeric Readout** capable of displaying one or two-digit octal, decimal, or hexadecimal counts; a **Shift Register** capable of shift-right serial-to-parallel, shift-left serial-to-parallel, or parallel-in parallel-out data operations; a **BCD Counter** which is a four-bit synchronous up/down binary-coded-decimal counter; a **Binary Counter** which is a four-bit ripple counter that provides divide-by-two, divide-by-eight (000 to 111 counter), and divide-by-sixteen (0000 to 1111 counter) circuits; a **Full Adder** that performs the addition of two four-bit binary numbers, produces a four-bit sum, has a carry-in, and a carry-out; a **Latch** containing four flip-flops with true and complement outputs, two clock inputs for two two-bit latches, or a four-bit latch with the clock inputs tied together; an **Exclusive OR** module composed of four dual-input Exclusive OR gates; and a **Universal Module** that is a 16-pin dual in-line integrated circuit socket with two inputs to each pin, six terminals at ground and +5V, and can therefore accommodate any standard 16-pin SSI, MSI, or LSI chip.

The optional modules offer flexibility and expediency when implementing simple or complex circuits with the basic modules of the digital trainer. They allow the instructor or interested student to extend the trainer hardware capability and circuit complexity, implement more advanced digital logic circuits with the convenience that TTL MSI logic affords, utilize TTL MSI chip configurations as they exist in the industry, minimize circuit set-up time whether assembling basic or advanced circuits, and design these more advanced and complex logic circuits limited only by their own imagination and expertise.

The **Dual Numeric Readout** module consists of six chips; three for each display digit. Each digit in the numeric readout is fed by a decoder-driver chip, which is fed by a hex-inverter chip, which is fed by the weighted binary inputs 1, 2, 4, and 8. The inputs are inverted so that a low at these inputs will activate the desired display segments; having no inputs, therefore, will produce a display of 00. The number displayed is the sum of the weighted binary inputs that have a low at their input. In other words, a low only at input 1 would cause a 1 to be displayed; a low at inputs 1 and 2 would cause the sum 3 to be displayed; and a low at inputs 1, 2, and 4 would cause a 7 to be displayed. Each digit of the numeric readout has a hexadecimal capability; in other words, each digit is capable of displaying the alphameric characters 0 through 9 and A, b, C, d, E, F. Consequently, depending on the sum

of the low inputs, the numeric readout can display any octal, decimal, or hexadecimal count produced by any circuit in the digital trainer.

A two-bit, three-bit, or four-bit binary count, for example, with the 1 input being the LSB, would cause a single numeric digit to display a decimal 0 to 3 count, an octal or decimal 0 to 7 count, or a hexadecimal 0 to F count, respectively. A four-bit BCD count would cause a single digit to display a decimal 0 through 9.

An eight-bit BCD count, with the four low-order BCD bits connected to the 1 (LSB), 2, 4, and 8 inputs of the right display digit and the four high-order BCD bits connected to the 1, 2, 4, and 8 (MSB) inputs of the left display digit, would cause the numeric readout to display a decimal 00 through 99.

An eight-bit binary count, with the four low-order bits connected to 1 (LSB), 2, 4, and 8 of the right display digit and the four high-order bits connected to 1, 2, 4, and 8 (MSB) of the left display digit, would cause the numeric readout to display a hexadecimal 00 through FF.

The **Shift Register** module is composed of four R-S master-slave flip-flops, four AND OR INVERT gates, one AND OR gate, and six inverter-drivers. The register can perform a shift-right serial-to-parallel operation, a shift-left serial-to-parallel operation, or a parallel-in parallel-out operation depending upon the logic level of MODE CON (mode control), some external pin connections, and the presence of parallel or serial inputs. The shift register can be cascaded for any of these operations.

When MODE CON is low, the serial data is applied to SER(in) and the data is shifted right by CLK_1 which was enabled by MODE CON being low; CLK_2 and parallel inputs A(in) through D(in) are inhibited. After the serial data is shifted right into the four flip-flops, it exists at the parallel outputs Q_A through Q_D.

When MODE CON is high, CLK_1 is inhibited and CLK_2 is enabled. This allows either a parallel-in parallel-out or shift-left serial-to-parallel operation. The parallel-in parallel-out operation requires that the input data be placed at inputs A(in) through D(in). CLK_2 loads the data into the flip-flops and the data then exists at the parallel outputs Q_A through Q_D. The shift-left serial-to-parallel operation requires that the outputs Q_B through Q_D be connected to the parallel input of the previous flip-flop; in other words, Q_D to C(in), Q_C to B(in), and Q_B to A(in) while entering serial data at D(in). CLK_1 and CLK_2 can be fed by the same source if both shift operations can be sequenced by the same clock. Information should be present at the A(in) through D(in) inputs before clocking. Transfer of data to the outputs Q_A through Q_D occurs on a high-to-low transition of the clock input.

This **BCD Counter** module is a synchronous binary-coded-decimal up/down counter consisting of four J-K master-slave flip-flops and control gating. Synchronous operation is provided by all flip-flops being clocked simultaneously so that the outputs change at the same time. The outputs of the four master-slave flip-flops are triggered on a low-to-high transition of the CLK (clock) input if ENBL (enable) is low; a high at ENBL inhibits counting. Level changes at ENBL should be made only when the CLK input is high. The count direction is determined by the state of DN/UP (down/up); when it is low, the counter counts up and when it is high, the counter counts down. The counter is presettable; in other words, the outputs may be preset to any state by placing a low on the LOAD input and then entering the desired count at A(in), B(in), C(in), and D(in). The output will change to agree with the data inputs independently of the state of the input CLK. The counter can therefore be used as a modulo-N counter by modifying the count length with the preset inputs A(in), B(in), C(in), and D(in) and will count up or down from this preset count. The BCD counters can be easily cascaded by feeding the RCLK (ripple clock) output to ENBL of the succeeding counter if parallel clocking is used; RCLK can be fed to CLK input if parallel enabling is used. RCLK produces a low level output pulse equal in width to the low level portion of the CLK input when a count overflow or underflow exists. The MAX MIN (maximum/minimum count) produces a high output pulse approximately equal in duration to one clock cycle when the BCD counter overflows or underflows. The MAX MIN count output can be used to accomplish look-ahead for high speed operation.

The **Binary Counter** module is a four-bit binary ripple counter consisting of four J-K master-slave flip-flops which are internally configured to provide a divide-by-two, a divide-by-eight (000 to 111 counter), or a divide-by-sixteen (0000 to 1111 counter) circuit. A gated counter reset line inhibits the count inputs and simultaneously resets the counter outputs to zero. The reset line is controlled by Ro(1) and Ro(2) which must both be a high to stop the count and reset the counter to zero. By using A(in) for the pulse input and either or both Ro(1) and Ro(2) being low, A(out) provides a divide-by-two output. By using B(in) for the pulse input and either or both Ro(1) and Ro(2) being low, D(out) provides a divide-by-eight output. B(out), C(out), and D(out) provide a binary count from 000 to 111 where B(out) is the LSB and D(out) is the MSB. By using A(in) for the pulse input, connecting A(out) to B(in), and either or both Ro(1) and Ro(2) being low, D(out) provides a divide-by-sixteen output. A(out), B(out), C(out), and D(out) provide a binary count from 0000 to 1111 where A(out) is the LSB and D(out) is the MSB.

The **Full Adder** module performs the addition of two four-bit binary numbers. Inputs A_1(in) through A_4(in) are one four-bit data word. Inputs B_1(in) through B_4(in) are a second four-bit data word. A_4 and B_4 are the MSB of each data word. A_1 and B_1 are the LSB of each data word. The outputs Σ_1 through Σ_4 are the sum (Σ) bits for A_1-B_1, A_2-B_2, A_3-B_3, and A_4-B_4, respectively. C_0 is the carry-in to the full adder. The input conditions at A_1-B_1, A_2-B_2, and C_0 are used to determine outputs Σ_1 and Σ_2 and the value of an internal carry. The value of the internal carry, A_3-B_3, and A_4-B_4 are then used to determine Σ_3, Σ_4, and C_4, the carry out.

The **Latch** module contains four D-type flip-flops with data inputs 1D(in) through 4D(in), data outputs 1Q through 4Q, and their complements $1\bar{Q}$ through $4\bar{Q}$, a CLK(1-2) input for the first two flip-flops, and a CLK(3-4) input for the second two flip-flops. For a four-bit latch, both clocks should be tied together so that the outputs 1Q through 4Q will become whatever the inputs 1D(in) through 4D(in) are at the clock high-to-low transition. The information present at the data inputs, 1D(in) through 4D(in), is transferred to the outputs, 1Q through 4Q, when the clock, CLK(1-2) and CLK(3-4), is high and the outputs will follow the data input as long as the clock remains high. When the clock goes low, the information that existed at the data inputs at the time the high-to-low clock transition occurred is retained at the outputs until the clock returns high. If CLK(1-2) and CLK(3-4) are not connected, two independent two-bit latches may be used.

The **Exclusive OR** module contains four dual-input Exclusive OR (XOR) gates. Each gate performs the function $Y = A\bar{B} + \bar{A}B$ or $Y = A \oplus B$. In other words, when the two inputs to an XOR gate are complementary (1 and 0), the output is a logic 1. If the inputs are the same (0 and 0, 1 and 1), the output is a logic 0.